『論理と形式化』資料 No.0
言語

亀山幸義 (kam[at]cs.tsukuba.ac.jp)

1　一階述語論理の言語

論理体系の「言語 (language)」とは、論理式を書くための「ことば」であり、一階述語論理に対する言語の場合は、以下の要素から構成される。

- 論理記号: $\bot, \wedge, \vee, \supset, \neg, \forall, \exists$ (その他に $\equiv$ なども使うことがある。)
- 変数: $x, y, z, \cdots$ (必要に応じて、x_1 のように添字を使う。以下、同じ。)
- 定数: $c, d, \cdots$
- 関数記号: $f, g, h, \cdots$ (各関数記号は、アリティが決まっている。)
- 述語記号: $P, Q, R, \cdots$ (各述語記号は、アリティが決まっている。)

このほか、開きかっこ、閉じかっこ などの補助記号を使うことがある。

関数記号や述語記号は、中置記法で表すことがある。アリティ(arity) は、引数の個数の意味であり、たとえば、$+$ (加算) は、引数を 2 つとって $a+b$ と書くので、アリティ2 (あるいは 2 引数) であるという。

上記の言語の構成要素のうち、、定数、関数記号、述語記号は、対象とする問題領域によって異なる記号を使う。

例 1: ペアノ算術の言語 L_1.

- 定数: 0.
- 関数記号: s (1 引数), $+$, $*$ (以上、2 引数).
- 述語記号: $=$ (2 引数).

$s(e)$ は「e に 1 を加えた自然数」を表す。たとえば、$s(s(0))$ は自然数の「2」を表す。

言語 L_1 の論理式の例:

- $x = s(0)$.
- $(x = s(y)) \vee (x = s(y * s(s(0))))$.
- $\forall x\ \exists y\ (x = s(s(0)) * y \vee x = s(s(s(0)) * y))$. (すべての自然数は偶数であるか、奇数であるかどちらかである。)

例 2: 言語 L_1 を拡張した言語 L_2.

- 定数: 0.
- 関数記号: s, ! (以上、1 引数), $+$, $*$, $-$, $/$ (以上、2 引数).
- 述語記号: $=$, $<$, $>$, $\leq$, $\geq$ (以上、2 引数).

言語 L_2 の論理式の例:

- $x!! \geq y$

- $\forall x\ (x \geq y \supset x = y \lor x > y)$.
- $\forall x\ \exists y\ (x < y \land \neg(\exists z\ \exists w\ (z > s(0) \land w > s(0) \land y = z * w)))$. (どんな自然数よりも、大きな素数が存在する。)

例 3: 親子、祖父母と孫の関係をあらわす言語 L_3.

- 定数: alice, bob, chris, emma.
- 関数記号: なし。
- 述語記号: isParent, isGrandParent (以上、2 引数).

言語 L_3 の論理式の例:

- isParent(alice,bob).
- $\forall x\ \forall y\ \forall z$ (isParent$(x,y) \land$ isParent$(y,z) \supset$ isGrandParent(x,z)). (親の親は、祖父母である。)

言語 L の上の論理式の構文を、BNF (Backus Normal Form) で定義する。

$$
\begin{aligned}
\text{項 } t &::= x \mid c \mid f(t, t, \cdots, t) \\
\text{原子論理式 } A &::= \perp \mid P(t, t, \cdots, t) \\
\text{論理式 } F &::= A \mid (F \land F) \mid (F \lor F) \mid (F \supset F) \mid (\neg F) \mid (\forall x\ F) \mid (\exists x\ F)
\end{aligned}
$$

ただし、項 $f(t_1, \cdots, t_n)$ は、f のアリティが n の時のみ構成でき、原子論理式 $P(t_1, \cdots, t_n)$ は、P のアリティが n の時のみ構成できるものとする。

例: 言語 L_1 では、$0 + x$ は項であるが、$+x$ は項ではない。

原子論理式において、0 引数の述語の場合は、$P()$ となるべきだが、括弧を省略して P と書くことがある。

論理式の括弧は適宜省略する。($\land, \lor, \supset, \neg, \forall, \exists$ の順に結合力が強いものとし、同じ記号の間では、$\land, \lor$ は左結合、$\supset$ は右結合とする。) たとえば、$P \land P \land P \supset P \supset P$ は、$((P \land P) \land P) \supset (P \supset P)$ と括弧付けできる。

2 形式的体系と意味論

形式的体系 (formal system) の世界と、意味論 (semantics) の世界は、完全に分離されていることを、しっかり認識しよう。

形式的体系		意味論
公理、推論規則	$\Longleftarrow$ 別の世界 $\Longrightarrow$	付値、解釈、構造、モデル
証明できる/できない ($\vdash A$)		正しい (真)/正しくない (偽) $\models A$)

2 つの世界は独立に定義され、あとで、両者が一致するとかしないかを議論する。(証明できる論理式の集合と、常に正しい論理式の集合は、命題論理や、一階述語論理の場合は、**たまたま**うまく作ったので一致する。)

例 1. 古典 命題論理: 形式的体系=NK, 意味論=真理値表による意味論
例 2. 直観主義 命題論理 (参考): 形式的体系=NJ, 意味論=クリプキモデル
例 3. 古典 一階述語論理: 形式的体系: NK の体系、意味論: 後述 (第 9 週の話題)。